NPS-69Zc76111

NAVAL POSTGRADUATE SCHOOL

Monterey, California

AN OPTIMAL COMPACT STORAGE SCHEME FOR
NONLINEAR REACTOR PROBLEMS BY FEM

by

D. Salinas, D. Nguyen and R. Franke

November 1976

NAVAL POSTGRADUATE SCHOOL
Monterey, California

Rear Admiral Isham Linder
Superintendent

J. R. Borsting
Provost

AN OPTIMAL COMPACT STORAGE SCHEME FOR NONLINEAR REACTOR PROBLEMS BY FEM

This work shows that optimal compact storage of coefficient matrices affords a significant reduction in core storage requirements over banded storage schemes. The resulting savings enables in core finite element solutions of large systems not otherwise possible. It is shown that Gears method for the stiff system of a nonlinear reactor dynamics problem is not as efficient as Crank-Nicolson integration because of substantially greater core requirement, despite its superior tracking ability. A remedy in the form of a modified implicit version of Gears method with a significant reduction in core requirement is shown to provide the same excellent accuracy as Gears method. Comparisons between the modified Gear method and the Crank-Nicolson method show the relative advantages and disadvantages of each. Finally, it is shown that although the nonlinearity encountered in this problem can be treated directly, a linear approximation of the nonlinear term affords a substantial reduction in core requirement with a relatively small cost in accuracy.

NPS-69Zc76111
November 1976

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NPS-69Zc76111	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* AN OPTIMAL COMPACT STORAGE SCHEME FOR NONLINEAR REACTOR PROBLEMS BY FEM		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) D. Salinas D. Nguyen R. Franke		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Code 69Zc, Mechanical Engineering Department Naval Postgraduate School Monterey, CA 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE November 1976 13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*		15. SECURITY CLASS. *(of this report)* UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

finite element, nonlinear, reactor dynamics, optimal compact storage.

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This work shows that optimal compact storage of coefficient matrices affords a significant reduction in core storage requirements over banded storage schemes. The resulting savings enables in core finite element solutions of large systems not otherwise possible. It is shown that Gears method for the stiff system of a nonlinear reactor dynamics problem is not as efficient as Crank-Nicolson integration because of substantially greater core requirements, despite its superior tracking ability. A remedy in the form of a modified implicit versior

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601 |

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

of Gears method with a significant reduction in core requirements is shown to provide the same excellent accuracy as Gears method. Comparisons between the modified Gear method and the Crank-Nicolson method show the relative advantages and disadvantages of each. Finally, it is shown that although the nonlinearity encountered in this problem can be treated directly, a linear approximation of the nonlinear term affords a substantial reduction in core requirement with a relatively small cost in accuracy.

AN OPTIMAL COMPACT STORAGE SCHEME FOR NONLINEAR REACTOR PROBLEMS BY FEM

by

D. Salinas, D. Nguyen and R. Franke

Abstract - This work shows that optimal compact storage of coefficient matrices affords a significant reduction in core storage requirements over banded storage schemes. The resulting savings enables in-core finite element solutions of large systems not otherwise possible. It is shown that Gears method for the stiff system of a nonlinear reactor dynamics problem is not as efficient as Crank-Nicolson integration because of substantially greater core requirement, despite its superior tracking ability. A remedy in the form of a modified implicit version of Gears method with a significant reduction in core requirements is shown to provide the same excellent accuracy as Gears method. Comparisons between the modified Gear method and the Crank-Nicolson method show the relative advantages and disadvantages of each. Finally, it is shown that although the nonlinearity encountered in this problem can be treated directly, a linear approximation of the nonlinear term affords a substantial reduction in core requirement with a relatively small cost in accuracy.

Statement of the Problem

The finite element method (FEM) has been successfully applied to the problem of a space-time nonlinear reactor with temperature dependent feedback (1,2). However, the effective numerical treatment of the resulting system of stiff O.D.E. called for further investigation. This work compares alternative computational schemes in a FEM solution of a nonlinear

nuclear reactor problem.

A test case is chosen which consists of a fully-reflected fast reactor, subject to super prompt critical conditions. By neglecting the delayed neutrons and using the prompt feedback model, the following nonlinear reactor dynamics equation is obtained in (r,z) space in the core,

$$\frac{1}{v}\frac{\partial \psi}{\partial t}(r,z,t) - D\Delta^2\psi + \lambda\Sigma_a\psi - w\psi^2 = 0 \qquad (1)$$

A similar equation without the nonlinear feedback term results in the reflector. A detailed development of these equations is given in (1).

In accordance with Galerkin's method, Eq. (1) is transformed into the implicit system of N ordinary differential equations,

$$\sum_{j=1}^{N} (A_{ij}\dot{\psi}_j(t) - B_{ij}\psi_j) + \sum_{j=1}^{N}\sum_{k=1}^{N} C_{ijk}\psi_j\psi_k = 0 \qquad (2)$$

$$i, = 1, \ldots, N$$

together with appropriate initial conditions, $\psi_i(0)$, i = 1, ..., N. The computational efficiency of any finite element program depends on the alternative schemes available for selection by the investigator. The present work considers the relative merits of various integration methods, algebraic equation solvers, array storage methods, as well as the effects of linearization.

Linearization

A well known procedure for treating nonlinear terms in FEM is by a "linearization" of the nonlinear terms. Linearization here refers to the process whereby a nonlinear term $F[\psi_i(t)]$ is approximated by $F[\psi_i(t-\Delta t)]$, where $\psi_i(t-\Delta t)$is the known value of ψ_i at the previous time. For the par-

icular quadratic nonlinear term $\sum_{j=1}^{N} \sum_{k=1}^{N} C_{ijk}\psi_j(t)\psi_k(t)$ of Eq. (2), a better linear approximation is

$$\sum_{j=1}^{N} C^*_{ij}\,\psi_j(t), \quad \text{where}$$

$$C^*_{ij} = \sum_{k=1}^{N} C_{ijk}\psi_k(t-\Delta t) \tag{3}$$

It should be noted that although the nonlinear quadratic term $\Sigma\Sigma C_{ijk}\psi_j\psi_k$ can be treated directly, and indeed it was in reference (1), the linear approximation, Eq. (3), results in a substantial reduction in both array storage and CPU time for a relatively small expense in accuracy, less than 5 percent for the cases run. The savings in storage and CPU time depends on the integration method and the algebraic equation solver employed, as well as the method of array storage.

By far the most common (but not optimal) method of array storage is the band storage scheme, whereby only the bandwidth (or half bandwidth in the case of symmetric matrices) terms in a matrix are stored. Since matrices obtained in FEM analyses are always banded, banded storage is always advantageous. Table 1 lists the storage requirements and the storage savings accomplished by linearization of symmetric and nonsymmetric C_{ijk}. Coefficient α is the number of bytes per word for the particular computer being utilized. For an IBM360 computer, α=4 for single precision, and 8 for double precision. N is the number of unknowns and n is the total bandwidth.

TABLE 1.

Condition \ Matrix	C_{ijk}	C^*_{ij}	Reduction
non sym., non banded	αN^3	αN^2	$\alpha N^2(N-1)$
non sym., banded	$\alpha N n^2$	$\alpha N n$	$\alpha N n(n-1)$
sym., non banded	$\frac{\alpha N}{2}(N+1)(\frac{1}{2} + \frac{2N+1}{6})$	$\frac{\alpha N(N+1)}{2}$	$\frac{\alpha(N+1)N}{2}(\frac{2N+1}{6} - \frac{1}{2})$
sym., banded	$\frac{\alpha N}{2}(n+1)(\frac{1}{2} + \frac{2n+1}{6})$	$\frac{\alpha N}{2}(n+1)$	$\frac{\alpha N}{2}(n+1)(\frac{2n+1}{6} - \frac{1}{2})$

It should be pointed out that some additional reduction can be obtained for symmetric matrices by discarding the $\frac{n(n-1)}{2}$ zeroes in the lower right triangle of the banded N x n C^*_{ij} matrix. In most cases n is small relative to N and the $\frac{n(n-1)}{2}$ additional reduction is not worth the effort.

Table 2 gives an idea of the savings achieved using banded storage for some typical regular rectangular grids as shown in Figure 1. The results are associated with the symmetric C_{ijk} and C^*_{ij} matrices of the non-linear reactor dynamics problem considered in this work, in which linear triangular elements were utilized with single precision on an IBM360 computer. (R-1) is the number of rows and (S-1) the number of columns in the rectangular grid, with S > R.

TABLE 2.

Grid	C_{ijk}	C^*_{ij}	Reduction
R=5 S=8 N=40 n=3	5,600	1,120	4,480
R=10 S=20 N=200 n=23	80,000	9,600	70,400
R=20 S=40 N=800 n=43	1,056,000	70,400	985,600
R=40 S=60 N=2400 n=83	11,424,000	403,200	11,020,800

Optimum Compact Storage of Coefficient Matrices

Since the interpolation function for the i^{th} unknown is non-zero only over those elements which contain i (i.e., the basic functions in FEM have local support only), the matrices resulting from FEM are both banded and sparse. The sparce nature arises from the fact that the unknowns surrounding the i^{th} unknown are not consecutively numbered. Common practice takes advantage of the banded nature of these matrices via a banded storage scheme, which stores an NxN matrix as an Nxn matrix where N is the number of unknowns and n is the total bandwidth of the matrix. Using banded storage, special care in the numbering of unknowns can affect a substantial decrease in n. As a simple example of this, consider the regular rectangular grid of Figure 1, having (R-1) rows of elements, and (S-1) columns of elements. For linear triangular elements, sequential vertical numbering results in n = 2R + 3, whereas sequential horizontal

numbering gives $n = 2S + 3$. The saving achieved by vertical numbering, assuming $R < S$, is $2N(S-R)$ words, or $2\alpha N(S-R)$ bytes, where α is the number of bytes per word. For symmetric matrices the previous numbers are halved.

These results may be generalized for triangular elements of order t. For the rectangular grid of Figure 1, the bandwidth for triangular elements with t^{th} order polynomial interpolation is

$$n = 6Rt - 4R - 2t + 5 \qquad (4)$$

for non symmetric matrices, and the half bandwidth is

$$n_s = 3Rt - 2R - t + 3 \qquad (5)$$

for symmetric matrices. Equations (4) and (5) are derived on the basis of condensing out the interior element nodes. Table 3 presents the evaluations of these formulas for linear, quadratic and cubic elements for the rectangular grid of Figure 1.

TABLE 3.

t	n	n_s
1	2R + 3	R + 2
2	8R + 1	4R + 1
3	14R - 1	7R

For banded systems of linear algebraic equations, direct methods of the Gaussian elimination type are most often utilized, being preferred over iterative methods such as successive over-relaxation (SOR). Depending on the particular application, there are points in favor of each method. In the case of linear equilibrium (elliptic) boundary value problems Gaussian elimination has the following advantages:

1. Direct methods allow the use of iterative refinement.
2. If more than one right hand side must be processed, the initial cost of decomposition must be expended only for the first solution.

For transient (parabolic or hyperbolic) initial-boundary value problems, or multi-step nonlinear equilibrium problems, SOR has the following advantages:

1. The large number of iterations for a solution at the first step is expended only once, succeeding solutions require far fewer iterations since a good estimate of the solution is then available.
2. No working space is required by SOR and therefore matrices may be compacted to maximum density.

It is this last point which suggests the replacement of the banded storage schemes by optimum compact storage (OCS) which will be described shortly. The argument in defense of direct methods that storage is becoming increasingly abundant is a spurious one which impedes investigation of larger systems, especially for three dimensional problems. It will be shown here that optimum compact storage of coefficient matrices leads to a substantial saving of core for large systems. A description of OCS

follows.

Accounts of OCS are given by Tewarson (3) and Gustavson (4). The underlying idea behind OCS is simply to store only the non zero terms of a coefficient matrix. The reduction in storage requirement achieved through OCS as compared to banded storage is illustrated by the regular rect angular grid shown in Figure 1. It is striking that regardless of R and S, the band of n terms surrounding any nodal point contains, at most, 7 non zero coefficients in the case of linear triangular elements. For the cases of quadratic and cubic elements the maximum number of non zero coefficients for any nodal point is 19 and 30 respectively. Hence the maximum storage requirements for the regular grid of Figure 1 for linear, quadratic and cubic triangular elements are $\alpha(7N)$, $\alpha(19N)$ and $\alpha(30N)$, respectively. The actual reduction is not simply the difference between banded core requirement and OCS core requirement for two reasons.

1. OCS storage can be achieved with less than $\alpha(\gamma N)$ where γ is the maximum number of non zero coefficients and α is the number of bytes per word.
2. Additional vector arrays are required for the implementation of OCS.

Although the following discussion is given in terms of the linear NxN C^*_{ij} array, the nonlinear NxNxN C_{ijk} array is treated in the same way. Indeed the direct treatment of the nonlinearity by OCS provided the most impressive reduction in core imaginable. An additional word on this will be given at the end of this section. It is well to keep in mind, however, that although direct treatment of the nonlinearity is esthetically pleasing, it is a questionable luxury in view of the excellent results obtained

by linearization.

The implementation of OCS requires two integer array vectors, say ISTART and NAME, and a vector of the non zero coefficients of C, which we shall call CC. The i^{th} integer entry in the (N+1)x1 ISTART vector is the number q_i, where $q_i = \sum_{j=1}^{i-1} p_i+1$, and p_i is the number of terms in the i^{th} equation, i.e., p_i is the number of nodes surrounding the i^{th} node. ISTART, then, is a pointer vector whose i^{th} term locates the initial position in the CC vector of the contributing coefficients to the i^{th} equation. The Mx1 NAME vector, where $M = \sum_{i=1}^{N} p_i$, is composed of N successive vector blocks of variable length p_i, i=1,...,N. The p_i integer numbers in the i^{th} block of NAME identify the p_i contributors to the i^{th} equation. The Mx1 CC vector contains the real non zero coefficients of the NxN matrix C, arranged in the same contiguous block arrangement as the NAME vector. The j^{th} term in the i^{th} block, CC (ISTART (I) + J-1) is coefficient C_{Ik} where K = NAME (ISTART (I) + J-1). To fix ideas the vectors associated with the simple grid shown in Figure 1 are

$$\text{ISTART} = \langle\ 1\quad 5\quad 10\quad 13\quad 18\quad 25\quad 30\quad 33\quad 38\quad 41\ \rangle$$

$$\text{NAME} = \langle\ 1\ 2\ 4\ 5|\ 2\ 3\ 1\ 6\ 5|\ .\ .\ .\ .\ .|\ 9\ 8\ 7\ \rangle$$

$$\text{CC} = \langle\ C_{11}\ C_{12}\ C_{14}\ C_{15}|\ C_{22}\ C_{23}\ C_{21}\ C_{26}\ C_{25}|\ .\ .\ .|C_{99}\ C_{98}\ C_{97}\rangle$$

In this case N=9 and M=40.

With the interior nodes of an element condensed out, the length (number of entries) of the NAME and CC vectors for the regular rectangular grid of Figure 1 with triangular elements of t^{th} order polynomial interpolation, is:

$$M = \sum_{i=1}^{N} p_i = RS\,(15t^2 - 6t - 2) + (R + S)\,(-14t^2 + 8t + 2) + (13t^2 - 10t - 1) \qquad (5)$$

The number of unknowns is

$$N = RS\,(3t - 2) + (R + S)\,(2 - 2t) + (t - 1) \qquad (6)$$

Table 4 gives the evaluation of Equations (5) and (6) for linear, quadratic and cubic triangular elements for the grid of Figure 1.

TABLE 4.

t	N	M
1	RS	7RS - 4 (R + S) + 2
2	4RS - 2 (R + S) + 1	46RS - 38 (R + S) + 31
3	7RS - 4 (R + S) + 2	115RS - 120 (R + S) + 86

Hence the core requirements for banded and compact storage are:

$$N_B = \alpha N n^* \qquad \begin{array}{l} n^* = n \text{ for non sym. matrices} \\ n^* = n/2 \text{ for sym. matrices} \end{array}$$

$$N_C = \alpha M + \beta\,(M + N + 1) \qquad (7)$$

respectively, where α and β are the number of bytes per word for real and integer numbers. For an IBM360, α is 4 for single precision, and 8 for double precision; β is 2 for integers less than 2^{15} and 4 for integers greater than 2^{15}. Comparative results between banded and OCS storage,

using the formulas from Table 2 and 3, for single precision with an IBM360 computer for symmetric operators on some typical RxS rectangular grids are given in Table 5.

TABLE 5.

	R = 10 S = 20			R = 20 S = 40			R = 40 S = 60		
t	1	2	3	1	2	3	1	2	3
N	200	741	1,282	800	3,081	5,362	2,400	9,401	16,402
M	1,282	8091	19,486	5,362	34,551	84,886	16,402	106,686	264,086
n_s	12	41	70	22	81	140	42	161	280
N_B	9,600	121,524	358,960	70,400	998,244	3,002,720	403,200	6,054,244	18.4×10^6
N_c	8,094	50,030	119,482	33,774	213,470	520,042	103,214	658,920	1,617,322
$N_B - N_c$	1,506	71,492	239,478	36,626	784,774	2,482,678	299,986	5,395,324	16.8×10^6
% Dif	-15.7	-58.5	-66.7	-52.0	-78.6	-82.7	-74.4	-89.1	-91.2

The notation in Table 5 is as follows:

t - order of polynomial interpolation

N - number of unknowns

M - number of terms in the NAME and CC vectors

n_s - bandwidth for the symmetric matrix

N_B - core required for banded storage, single precision IBM360

N_C - core required for OCS, single precision IBM360

N_B-N_C - reduction in core storage by OCS, single precision IBM360

% Diff - percent difference between banded storage and OCS

In Table 5, there is no intention to suggest comparisons for a particular grid with respect to the order of polynomial interpolation. Comparisons with respect to N, the number of unknowns, are more meaningful. ISTART and NAME vectors are the same for the A and B matrices and hence the comparisons given in Table 5 are conservative for systems with more than one coefficient matrix.

The algorithm to assemble the element matrices into the CC vector is straight forward and can be accomplished in a dozen or so steps. In our work it was found useful to construct and have the computer punch out the ISTART and NAME vectors from the element-system connectivity of the discretized model under consideration. Again this pre-processing algorithm was quite simple to program.

Solution Methods

It was not the intention here to compare the many techniques available for the solution of Eq. (2), but rather to establish the computational features of the multi-step, predictor-corrector Gear method for stiff systems, and the single-step, implicit Crank-Nicolson method. The work led to the development of a modified implicit form of Gears method which incorporated some of the attractive features of each system.

Having been specifically developed for stiff systems, Gears method handled Eq. (2) quite efficiently with respect to CPU time. Indeed, Gears method achieved a specified accuracy criterion with far fewer integration

steps than the Crank-Nicolson method, which experienced particular difficulty during the early transient period of time. However, Crank-Nicolson performed equitably with Gears method during the later transient period as shown in Figure 2.

A severe drawback to multi-step methods such as Gears method, in contrast to single-step methods such as Crank-Nicolson, is the need to retain previous time solutions. The predictor-corrector Gear method, as typified in the DVOGER subroutine (5) is capable of 6^{th} order interpolation in time and requires N(N + 17) words of work storage, in addition to 8N words for storage of past history, unquestionably a considerable expense.

Another disadvantage of Gears method is the need to transform the implicit system of Eq. (2) into the explicit system

$$\dot{\psi} = A^{-1} (B - C\psi)\psi = F(\psi) \tag{8}$$

in the case of direct treatment of the nonlinear term, or

$$\dot{\psi} = A^{-1} (B - C^*)\psi \tag{9}$$

when the nonlinear term is given the linear approximation of Eq. (3). Equations (8) and (9) are formal statements since actual pre-multiplication by A^{-1} is not necessary. These equations may be obtained by a Cholesky decomposition into

$$LL^T\dot{\psi} = B\psi - C\psi^2 \tag{10}$$

or,

$$LL^T\dot{\psi} = (B - C)\psi \tag{11}$$

which, in turn, may be solved by forward and backward substitutions for $\dot{\psi}$. It should be noted that Cholesky decomposition eliminates the possible use of OCS, and hence a severe penalty in coefficient array storage is

incurred for the A matrix. This, in addition to the need to store the NxN Jacobian matrix $\frac{\partial Fi}{\partial \psi_j}$ gives a great advantage to the Crank-Nicolson method, despite the superior ability of Gears method for stiff systems. The modified Gears method, to be described shortly, offers a substantial diminishment in the storage requirement of Gears method, while maintaining the advantage of superior integration for stiff systems. The net effect is a method which compares favorably with the Crank-Nicolson method.

The Modified Gear Method

Our initial work (1) presented a dilemna which required additional investigation. On the one hand the superior tracking ability of Gears method for the stiff system of equations (2) was admired, while on the other hand there was dissatisfaction with the large core storage requirement. One way to minimize this difficulty was a modification of Gears method.

1. To treat the implicit system of Eq. (2) and thus achieve the core saving advantages of OCS, and
2. Eliminate the need for storing the NxN Jacobian matrix.

In order to obtain an understanding of the modified Gear method the following presentation is given.

The core storage requirements of the multi-step Gear method arise primarily from the necessity to compute and store the NxN Jacobian matrix $\partial Fi/\partial \psi_j$. While A, B and C (or C*) are sparse banded matrices, the Jacobian A^{-1} (B-2Cψ), or A^{-1}(B-C*), is full. Storage of this matrix accounts for N^2 of the work storage words required by DVOGER. At the

suggestion of Prof. C. W. Gear (6), an existing program (7) for implicitly defined stiff differential equations was modified to take maximum advantage of the sparse nature of matrices A, B, and C (or C*). This includes the use of OCS for the equivalent of the Jacobian matrix, which uses the same ISTART and NAME vectors as the A and B matrices, and the use of iteration (SOR) to solve for the difference between the predicted value and the corrected value at each time increment.

A brief background concerning Gears method for implicit stiff equations is necessary in order to understand why SOR results in a substantial increase in computational efficiency for problems of this type. In the general case, one seeks to obtain a numerical solution of the system of equations

$$F\ (t,\ y,\ \dot{y}) = 0 \tag{12}$$

with initial conditions $y(t_o)$ specified. At each time step here denoted by superscript k, the corrector equations

$$F\ (t,\ y^{(k)},\ -\alpha_o y^{(k)}/\beta_o h + \Sigma_k) = 0 \tag{13}$$

which result from approximating $\dot{y}$ by a linear combination of $y^{(k)}$ and previous solution values, must be solved for $y^{(k)}$. In the above, h is the stepsize, α_o and β_o depend on the order of the method (interpolation on time), and Σ_k depends on the order and previous solution values. Gears method uses Newtons method to solve for $y^{(k)}$, hence the Jacobian, $J = \frac{\partial F}{\partial y} - (\frac{\alpha_o}{\beta_o h}) \frac{\partial F}{\partial \dot{y}}$, (or a reasonable approximation to it) must be computed. Because the predictor provides a good estimate of $y^{(k)}$ and Newton's method solves for the difference in iterates, δy, the linear equations to be solved are of the form

$$J(\delta y) = -F. \tag{14}$$

When the solution δy is added to the predicted value it is apparent that δy need be accurate to only a few significant digits. Also, because up to three Newton iterations are performed, inaccuracies in the first iteration, where the largest corrections, δy, are expected, are automatically accounted for in succeeding iterations. It should be noted that the iteration used by Gear is actually a modifed Newton method since J is not evaluated at each iteration, nor even at each time increment.

The error tolerance in solving for δy by SOR was satisfied when either of two criteria were met

1. The difference in components of successive iterates were as small compared to the solution δy as that allowed by Gears method for the relative error in $y^{(k)}$, or
2. The difference in components of successive iterates compared to $y^{(k)}$ were less than 1 percent of the error allowed by Gears method in $y^{(k)}$.

Considering the comparitive magnitudes of the vectors $y^{(k)}$ and ∂y, the result was that very few SOR iterations are required to solve equation (14). In the problem under investigation here, 3-4 iterations were required for the first Newton iterate, and 1-2 for second and (if required) third Newton iterates.

A comparison of the number of arithmetic operations required for solution of the corrector equation (13) for three different schemes follows. An operation is defined to be a multiplication or division and an addition or subtraction.

1. Use of SOR in the modified Gear method results in M + 6N operations per iteration of SOR, including those required for

convergence tests. An overhead of N divisions for the initial estimate of the solution (taken to be $\delta y_i / J_{ii}$) is required for each Newton iteration. Storage for J requires only M locations for the coefficients since the ISTART and NAME vectors are the same as for the A and B matrices.

2. Use of Cholesky decomposition to factor J and solution of equations (14) by forward and backward substitutions requires $2Nn - n(n-1)$ operations for each Newton iteration, plus an overhead of about Nn^2 operations for Cholesky decomposition each time J is computed. While J is only recomputed every few time-steps, for large problems the decomposition of J will probably take as many operations as all the succeeding solutions of equation (14). The matrix J is assumed to be stored in symmetric band storage form, requiring Nn storage locations.
3. In a routine such as DVOGER where the explicit differential equations must be used, the banded nature of the Jacobian is destroyed and the resulting J matrix is full and nonsymmetric. Gauss elimination for the solution of (14) requires N^2 operations for forward and backward substitutions, with the LU decomposition requiring about $N^3/3$ operations each time J is computed. Since J is full, N^2 storage locations are required for it.

The total amount of work involved in solving the corrector equation (13) is difficult to compare in the first two instances because of a varying number of iterations required in solving (14) by SOR, as well as the variable nature of M compared to Nn. It is clear that DVOGER and

similar codes which must have the explicit equation, i.e., (8) or (9), to solve, cannot compare favorably with the modified Gear method which attacks the implicit equation (2), since both storage and run time requirements are greater. For the problems under consideration here, typical test runs indicated the solution of (14) by SOR is much faster than elimination, since the total run time for solution of the differential equation was less by a factor of about 2.5. With regard to storage, the advantage of SOR over Cholesky decomposition is greater than shown in Table 5 since the ISTART and NAME vectors are the same as for the A and B matrices. The details are presented in (8).

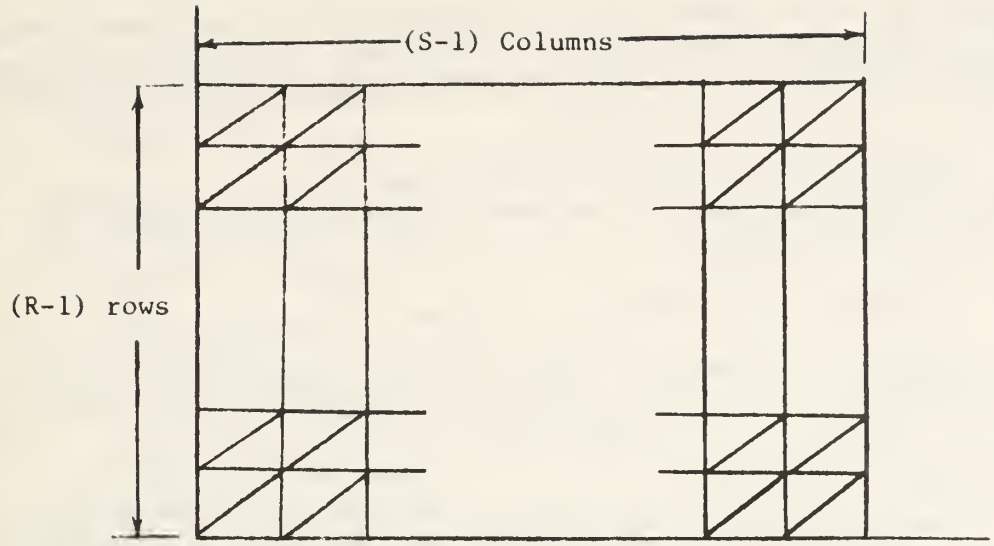

Figure 1. Rectangular Domain with Triangular Elements (S > R).

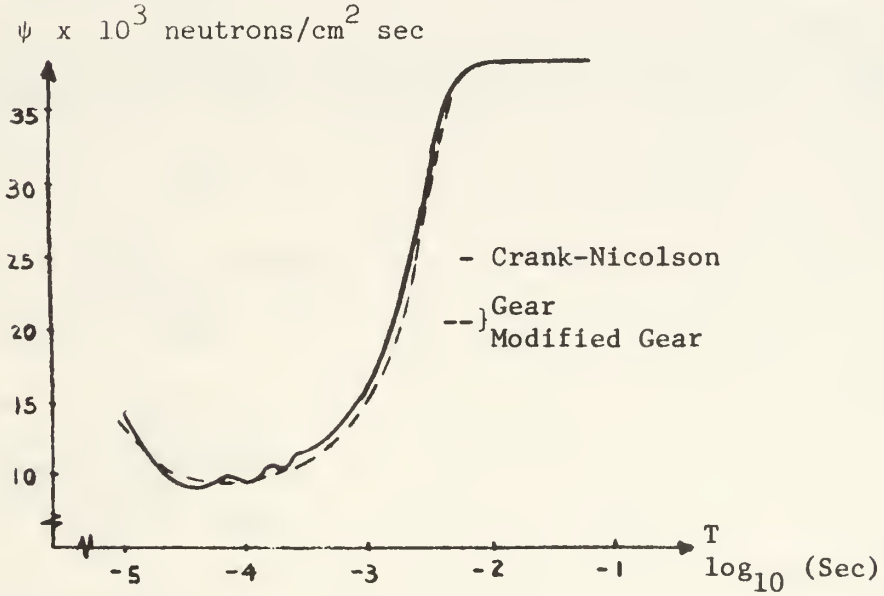

Figure 2. Plot of Center Point Neutron Flux for a Central Disturbance.

References

1. D. Salinas, D. H. Nguyen, and T. Southworth, "Finite Element Solutions of a Nonlinear Nuclear Reactor Dynamics Problem," in Computational Methods in Nonlinear Mechanics, J.T. Oden, Ed., The Texas Institute for Computational Mechanics, 1974, pp. 541-550.

2. D. H. Nguyen and D. Salinas, "Finite Element Solutions of Space-Time Nonlinear Reactor Dynamics," in Nuclear Science and Engineering, Vol. 60, No. 2, June 1976, pp. 120-130.

3. R. P. Tewarson, "Space Matrices," Academic Press, New York, 1973, Chapt. 1.

4. F. G. Gustavson, "Some Basic Techniques for Solving Sparse Systems of Linear Equations," in Sparse Matrices and Their Applications, D. J. Rose and R. A. Willoughby, Eds., Plenum Press, New York-London, 1972, pp. 41-52.

5. C. W. Gear, "The Automatic Integration of Ordinary Differential Equations," in Communications of the ACM, Vol. 14, No. 3, 1971, pp. 176-179.

6. Private Communication.

7. R. L. Brown and C. W. Gear, "Documentation for DFASUB - a program for the solution of simultaneous implicit differential and nonlinear equations," Report No. UIUCDCS-R-73-575, University of Illinois at Urbana-Champeigne, Urbana, Illinois, July 1973.

8. R. Franke, "A Program for the Numerical Solution of Large Sparse Systems of Algebraic and Implicitely Defined Stiff Differential Equations in NPS-53Fe76051, May 1976.

Distribution List

	No. Copies
Defense Documentation Center Cameron Station Alexandria, VA 22314	12
Library Naval Postgraduate School Monterey, CA 93940	2
Dean of Research Naval Postgraduate School Monterey, CA 93940	2
Naval Postgraduate School Department of Mathematics Monterey, CA 93940	
Dr. Carroll Wilde, Chairman, Code 53Wm	1
Dr. C. Comstock, Code 53Cs	1
Dr. F. Faulkner, Code 53Fa	1
Dr. R. Franke, Code 53Fe	5
Mr. R.S. Stepleman David Sarnoff Research Center Princeton, NJ 08540	1
Mr. R.L. Coleman The Charles Stark Draper Laboratory, Inc. 8 Albany Street Cambridge, MA 02139	1
Mr. R.I. Shrager National Institute of Health 9000 Rockville Pike Bethseda, MD 20014	1
Professor J. Dauer University of Nebraska Department of Mathematics 810 Oldfather Hall Lincoln, Nebraska 68508	1
Professor M. Karon Department of Mathematics Colorado College Colorado Springs, CO 80903	1
Naval Postgraduate School Department of Aeronautics Monterey, CA 93940	
Dr. D. Collins, Code 67Co	1
Dr. R. Ball, Code 67Bp	1

Distribution List Continued

	No. Copies
Naval Postgraduate School Department of Mechanical Engineering Monterey, CA 93940	
Dr. Allen E. Fuhs, Chairman, Code 69Fu	1
Dr. J.E. Brock, Code 69Bc	1
Dr. G. Cantin, Code 69Ci	1
Dr. R.E. Newton, Code 69Ne	1
Dr. D.H. Nguyen, Code 69Ng	5
Dr. D. Salinas, Code 69Zc	5
Naval Postgraduate School Computer Center Monterey, CA 93940	
Dr. D. Williams, Code 0141	1
Mr. Roger Hilleary, Code 0141	1
Professor D.A. Archer University of North Carolina Department of Mathematics Charlotte, NC 28213	1
Professor R.E. Barnhill Department of Mathematics University of Utah Salt Lake City, UT 84112	1
Professor M. Nielson Department of Mathematics Arizona State University Temper, AZ 85281	1
Mr. W.J. Dyka Naval Electronics Laboratory Center Code 4000 San Diego, CA 92152	1
Dr. Richard Lau Office of Naval Research Pasadena, CA 91100	1
Professor J.E. Taylor Aerospace Engineering and Applied Sciences University of Michigan Ann Arbor, Michigan 48104	1
Dr. M.Y. Ito Geophysics International Los Angeles, CA 90053	1

Distribution List Continued

	No. Copies
University of California at Los Angeles School of Engineering and Applied Science Los Angeles, CA 90024	
Dr. S.B. Dong	1
Dr. L. Selna	1
Dr. T.H. Lin	1
Dr. M. Rubinstein	1
Dr. R.B. Nelson	1
Dr. L. Felton	1
Dr. D. Konishi Rockwell International Los Angeles, CA 90053	1
Dr. Evan Weiner Hanford Engineering Development Laboratory Richland, Washington 99352	1
Dr. A. Galt NOAA Pacific Marine Environmental Laboratory University of Washington WB-10 Seattle, WA 98105	1
Professor J.S. Modeer Department of Mathematics University of Colorado Colorado Springs, CO 80907	1
Dr. G.K. Gupta Department of Computer Science Monarch University Clayton, Victoria 3168 Australia	1